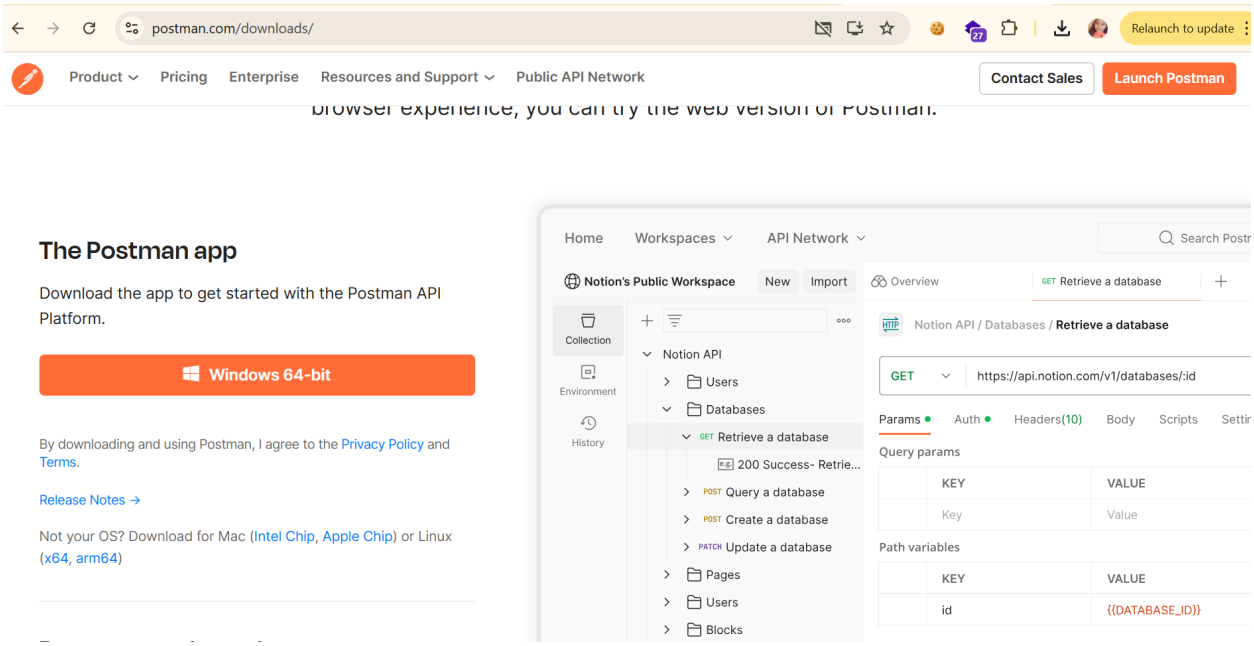
Session - 2 : Advanced Postman Techniques



POSTMAN

Join Postman

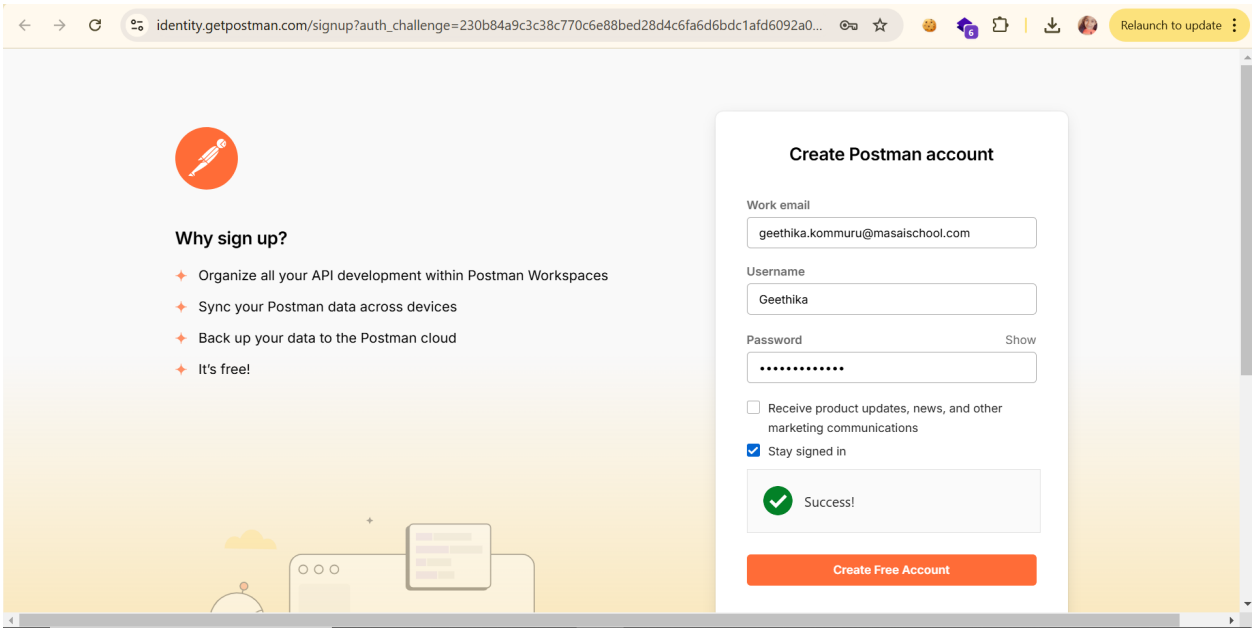
Work email

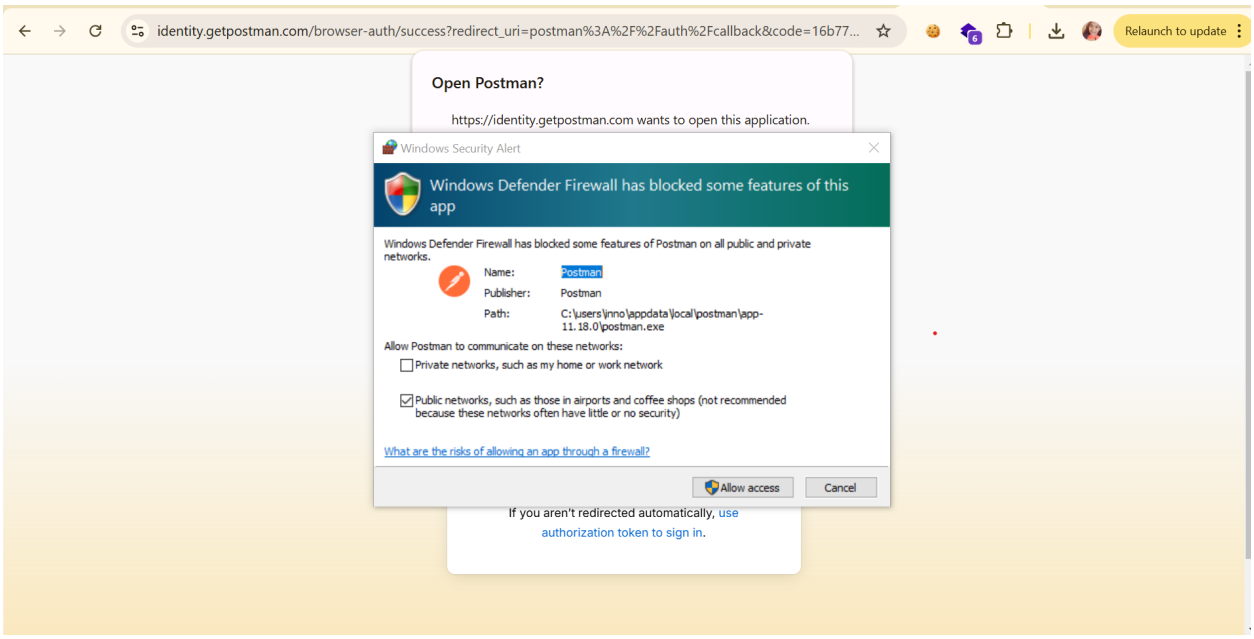
name@company.com

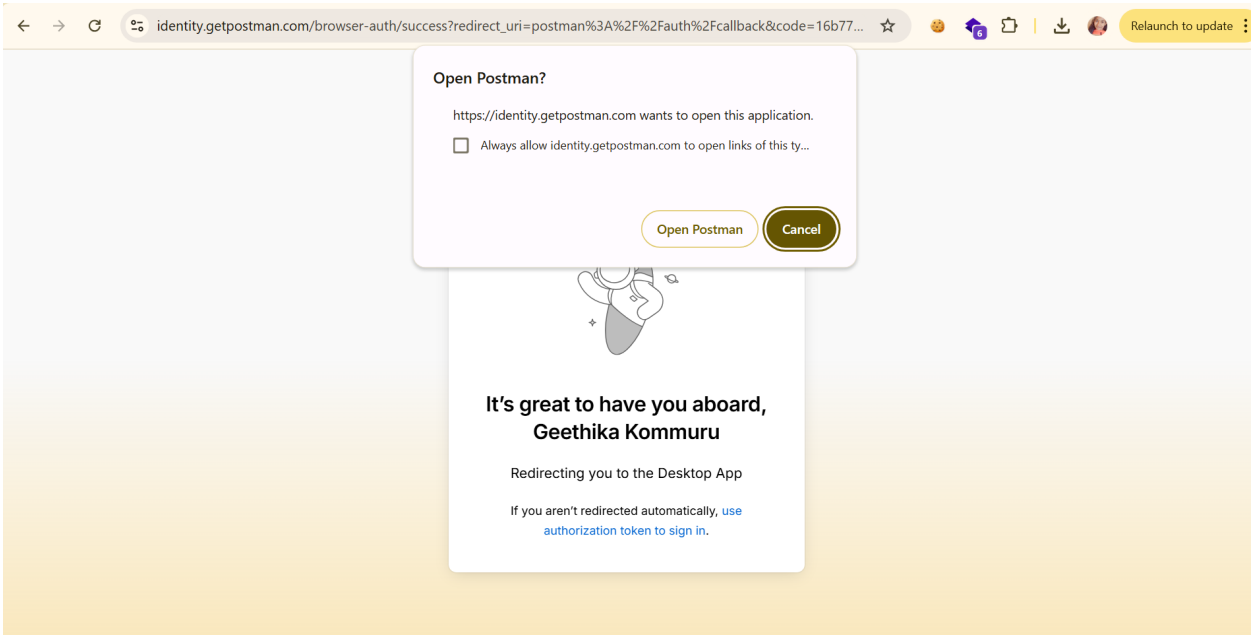
Create Free Account

Already have an account? Log In

Continue without an account



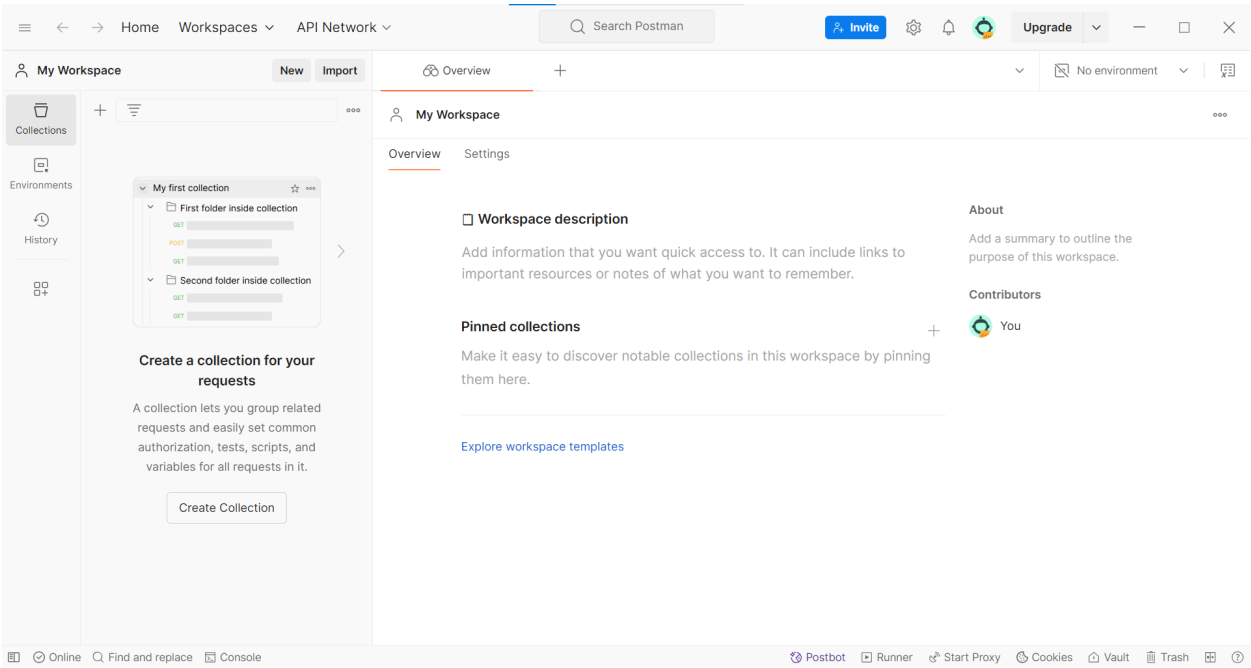


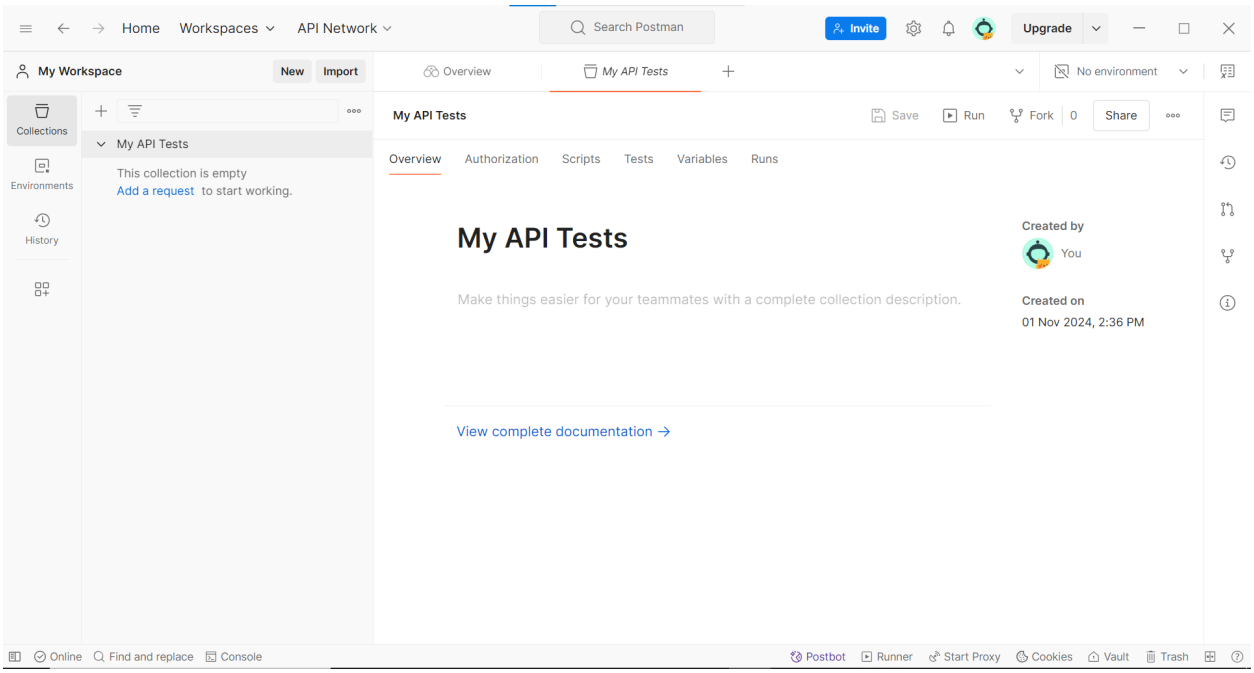


Step 1: Create a Collection

1. **Open Postman**.
2. In the left sidebar, click on **"Collections"**.

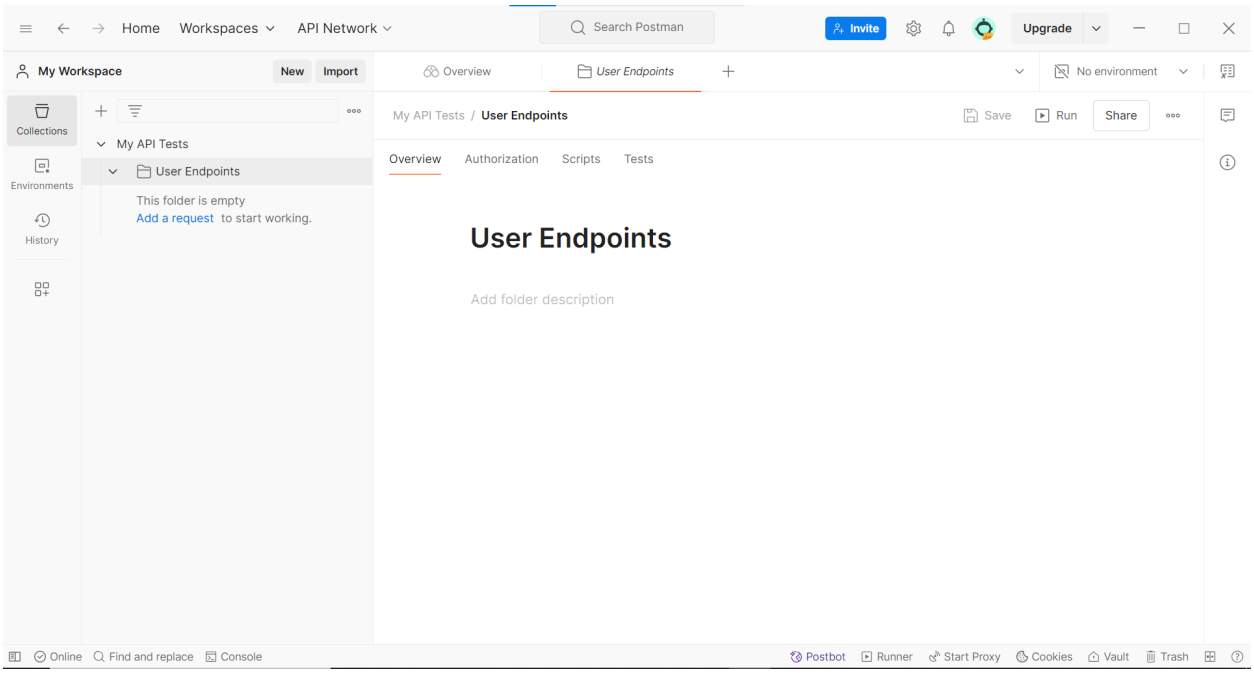
3. Click the **"+"** icon next to "Collections" or click **"New"** and select **"Collection"**.
4. **Name Your Collection**: For example, you can name it **"My API Tests"**.
5. (Optional) Add a description or any additional settings.
6. Click **"Create"** or **"Save"**.





Step 2: Create a Folder within the Collection

1. Navigate to your newly created collection (**"My API Tests"**).
2. Click on the three dots (•••) next to the collection name.
3. Select **"Add Folder"**.
4. **Name Your Folder**: For example, you can name it **"User Endpoints"**.
5. (Optional) Add a description for the folder.
6. Click **"Create"** or **"Save"**.



Here's how to create different types of requests (GET, POST, PUT, PATCH, DELETE) in Postman:

Step 1: Create a GET Request

1. **Open Your Collection**: Navigate to your collection (e.g., "My API Tests").
2. **Click on the Three Dots (•••)**: Next to your collection name, click the three dots.
3. Select **"Add Request"**.
4. **Name Your Request**: Name it something like **"Get Users"**.
5. Set the request type to **GET** using the dropdown menu next to the URL bar.
6. Enter the URL for your API endpoint (e.g., `https://api.example.com/users`).
7. Click **"Save"**.

Step 2: Create a POST Request

1. In the same collection, click the three dots again and select **"Add Request"**.
2. Name your request (e.g., **"Create User"**).

3. Set the request type to **POST**.
4. Enter the URL for the endpoint (e.g., `https://api.example.com/users`).
5. Click on the **"Body"** tab and choose **"raw"** and set the format to **JSON**.
6. Enter your JSON payload (e.g., `{"name": "John Doe", "email": "john@example.com"}`).
7. Click **"Save"**.

Step 3: Create a PUT Request

1. Click the three dots and select **"Add Request"** again.
2. Name your request (e.g., **"Update User"**).
3. Set the request type to **PUT**.
4. Enter the URL (e.g., `https://api.example.com/users/1`).
5. In the **"Body"** tab, select **"raw"** and set it to **JSON**.
6. Enter the updated JSON payload (e.g., `{"name": "Jane Doe", "email": "jane@example.com"}`).
7. Click **"Save"**.

Step 4: Create a PATCH Request

1. Click the three dots and select **"Add Request"**.
2. Name your request (e.g., **"Patch User"**).
3. Set the request type to **PATCH**.
4. Enter the URL (e.g., `https://api.example.com/users/1`).
5. In the **"Body"** tab, select **"raw"** and set it to **JSON**.
6. Enter the JSON payload with the fields to be updated (e.g., `{"email": "jane.doe@example.com"}`).
7. Click **"Save"**.

Step 5: Create a DELETE Request

1. Click the three dots and select **"Add Request"**.

2. Name your request (e.g., **"Delete User"**).
3. Set the request type to **DELETE**.
4. Enter the URL (e.g., `https://api.example.com/users/1`).
5. Click **"Save"**.

To create a global variable in Postman, follow these steps:

Step 1: Open the Manage Environments Window

1. **Open Postman**.
2. In the top right corner, click on the **gear icon** (⚙️) to open the "Manage Environments" window.

Step 2: Create a New Environment

1. Click on the **"Add"** button.
2. **Name Your Environment**: Enter a name for your environment (e.g., "Development").

Step 3: Add Global Variables

1. In the environment settings, you can add variables in the **"Variable"** section:
   - Enter the **Name** of the variable (e.g., `baseUrl`).
   - Enter the **Value** (e.g., `https://api.example.com`).
   - (Optional) You can add a **Description** for clarity.
2. Click on the **"Add"** button to create the variable.

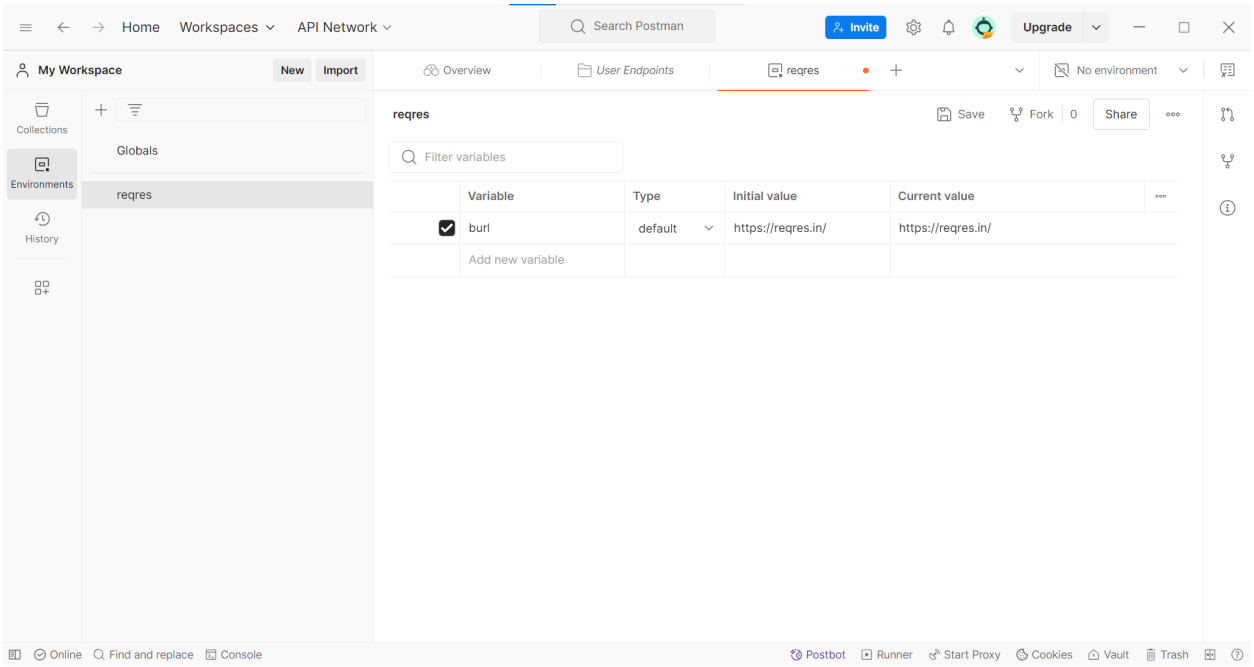
Step 4: Save the Environment

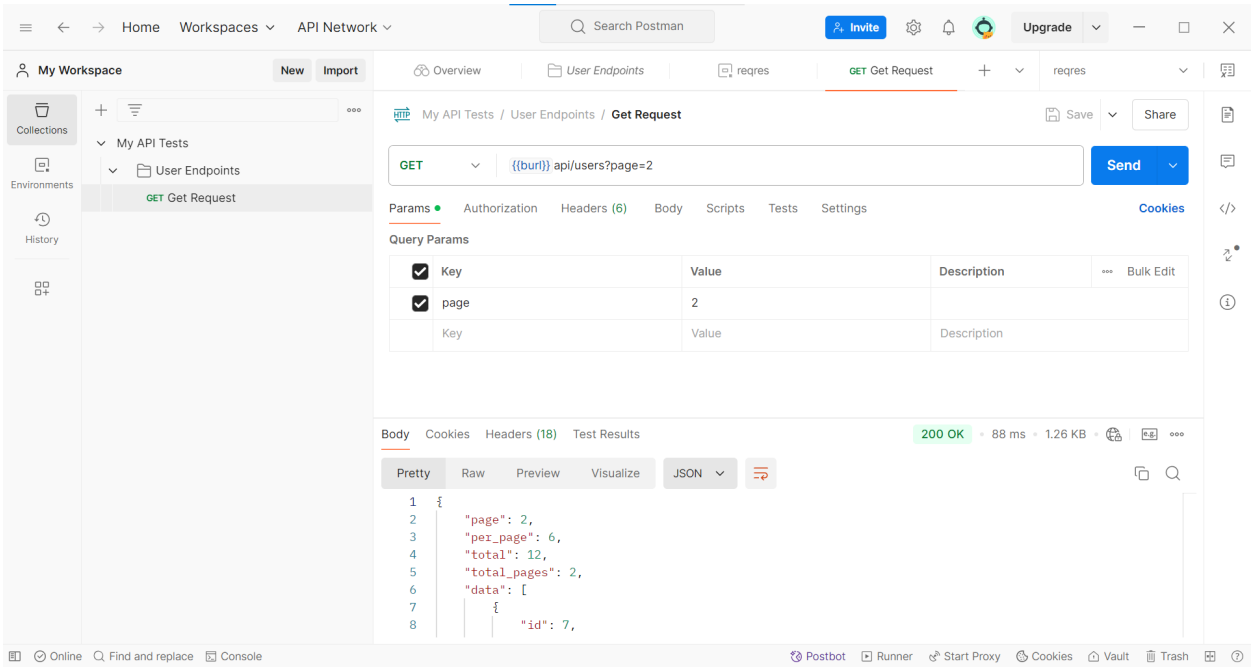
1. After adding all your desired variables, click **"Save"** to save the environment.

Step 5: Use Global Variables

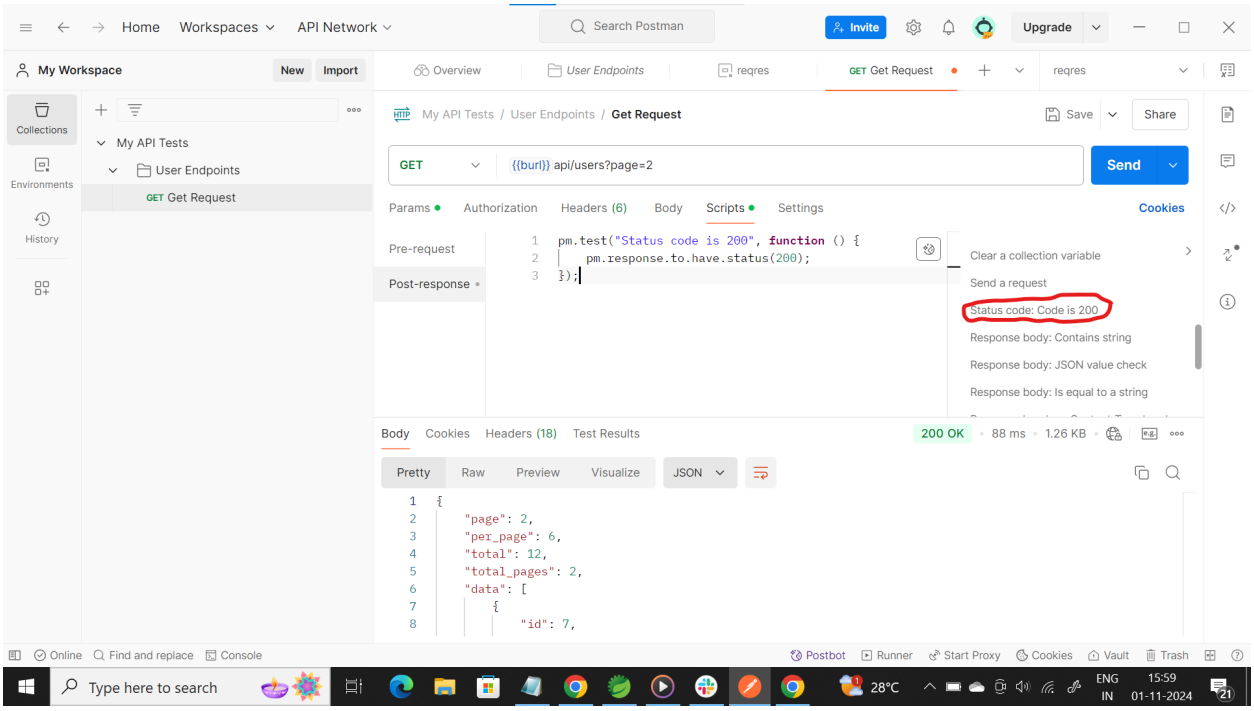
1. To use the global variable in your requests, you can reference it using the syntax `{{variableName}}`. For example, if you set `baseUrl` as a variable, you can

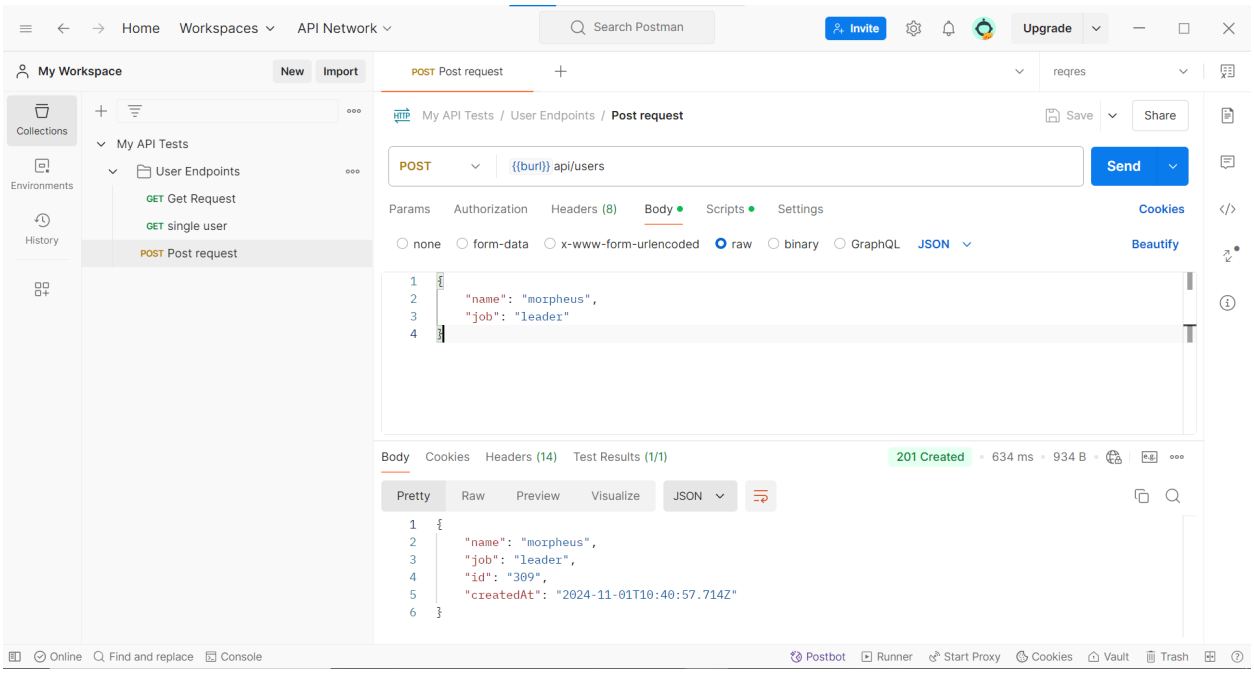
use `{{baseUrl}}/users` in your request URL.





Writing assertions :





SIMPLEBOOKS API :

> https://github.com/vdespa/introduction-to-postman-course/blob/main/simple-books-api.md

Step 1: Create a POST Request

1. **Open Postman**.
2. In your collection, click on the three dots (•••) next to it and select **"Add Request"**.
3. **Name Your Request**: For example, name it **"Create API Client"**.
4. Set the request type to **POST** using the dropdown next to the URL field.
5. Enter the URL: `https://simple-books-api.glitch.me/api-clients/`.

Step 2: Add the Request Body

1. Click on the **"Body"** tab.
2. Select **"raw"** and then choose **"JSON"** from the dropdown on the right.
3. Enter the following JSON in the body:(Replace `"Your Client Name"` and `"your.email@example.com"` with your actual data.)

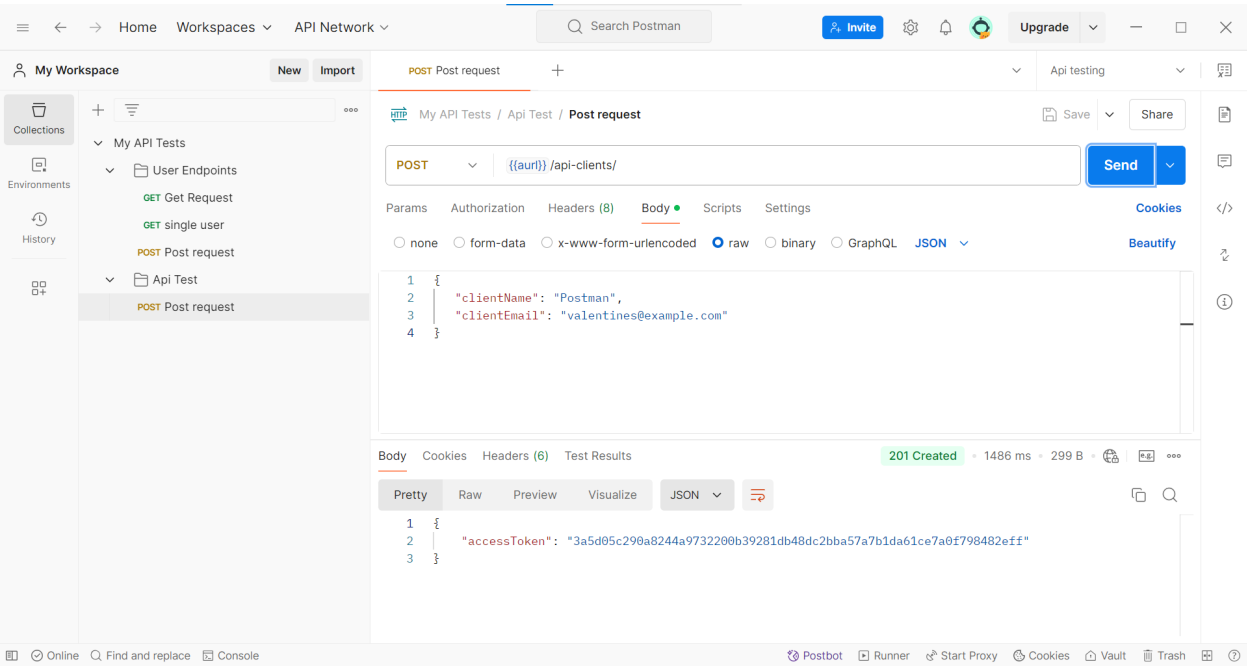
```
{
   "clientName": "Postman",
   "clientEmail": "valentin@example.com"
}
```

Step 3: Send the Request

1. Click the **"Send"** button.
2. Check the response section below for the access token. It should be included in the response.

Step 4: Copy the Access Token

1. In the response body, look for the access token (it might be labeled as `accessToken` or similar).
2. Select and copy the access token to use it in future requests.



Step 1: Create a POST Request

1. **Open Postman**.
2. In your collection, click on the three dots (•••) next to it and select **"Add Request"**.
3. **Name Your Request**: For example, name it **"Create API Client"**.
4. Set the request type to **POST** using the dropdown next to the URL field.

5. Enter the URL: `https://simple-books-api.glitch.me/orders`

Step 2: Add the Request Body

1. Click on the **"Body"** tab.
2. Select **"raw"** and then choose **"JSON"** from the dropdown on the right.
3. Enter the following JSON in the body:(Replace `"1"` and `"`John`"` with your actual data.)

```
{
  "bookId": 1,
  "customerName": "John"
}
```

Authorization :

Step 1: Obtain Your Bearer Token

Ensure you have copied the Bearer token from the response of your previous POST request.

Step 2: Open Your Request

1. Open the request where you want to add the Bearer token (e.g., the request you created for the API client).

Step 3: Go to the Authorization Tab

1. Click on the **"Authorization"** tab located below the request URL.

Step 4: Select Bearer Token

1. In the "Type" dropdown menu, select **"Bearer Token"**.

Step 5: Enter the Token

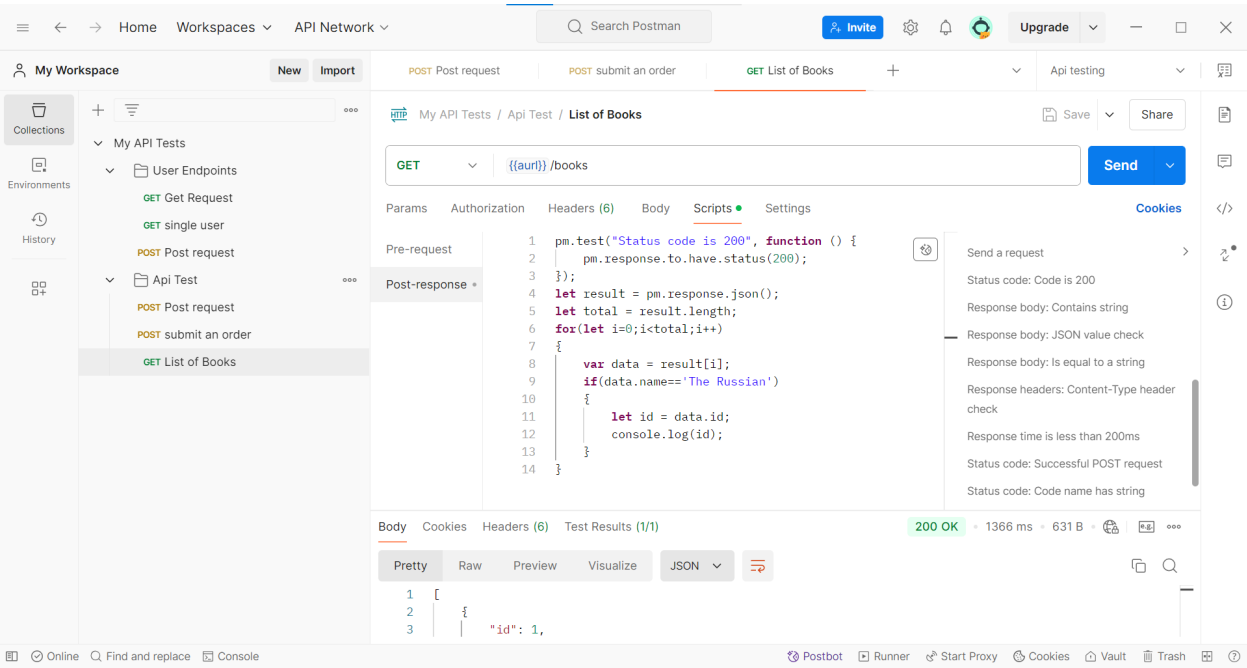
1. Paste your copied token into the **"Token"** field.

Step 6: Save Your Changes

1. Click on the **"Save"** button to save your request with the Bearer token.

Step 3: Send the Request

1. Click the **"Send"** button.
2. Check the response section below for the access token. It should be included in the response.




```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
let result = pm.response.json();
let res = result.accessToken;
console.log(res);
pm.environment.set("my_access", res);
```

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
let result = pm.response.json();
let total = result.length;
for(let i=0;i<total;i++)
{
    var data = result[i];
    if(data.name=='The Russian')
    {
        let id = data.id;
        console.log(id);

        pm.environment.set("my_id", id);
    }
}
```

Step 1: Open the Postman Console

1. In Postman, click on the **Console** icon located at the bottom left of the window (or press `Cmd + Alt + C` on macOS or `Ctrl + Alt + C` on Windows).

Step 2: Send the Request

1. Send the request that should return the ID you want to check.
2. After sending the request, observe the console for any output related to the ID. You may see logs printed here, including responses or any custom logs you added in the scripts.

Step 3: Set an Environment Variable (if applicable)

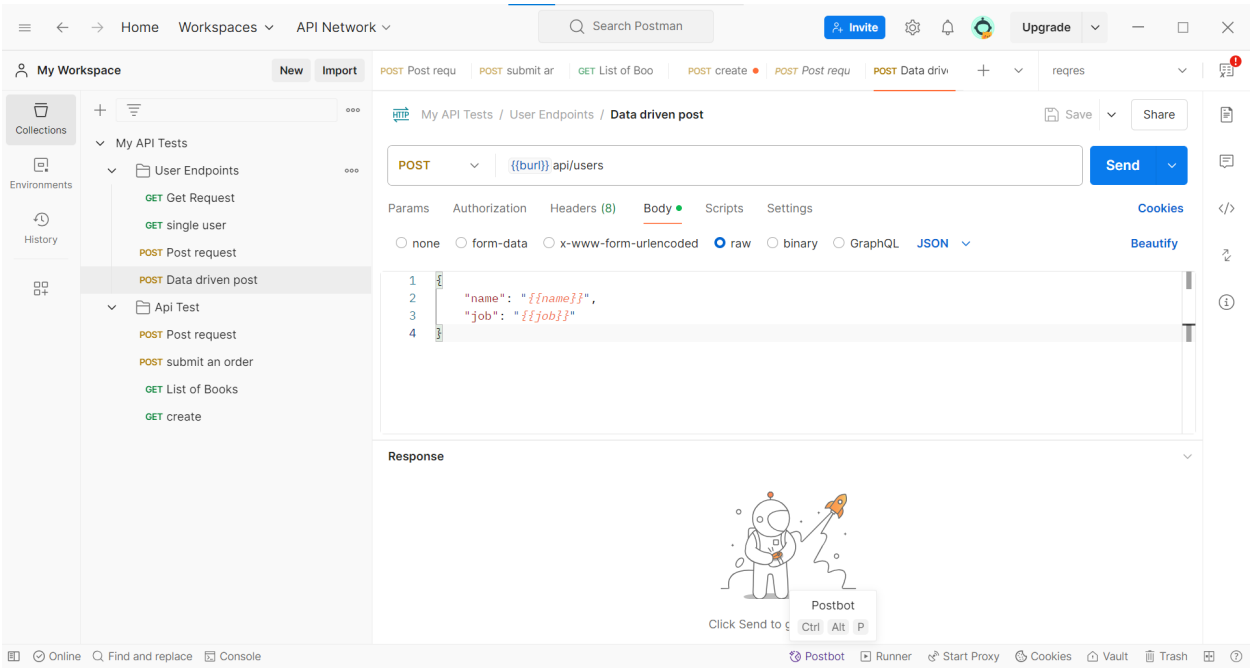
If you want to store the ID as an environment variable for later use, you can do the following:

1. In your request, go to the **"Tests"** tab.

Step 4: Use the Environment Variable

To use the environment variable in another request, you can reference it as `{{myId}}` in your request URL or body.

Data Driven:



To create a CSV file and run it as a collection in Postman, follow these steps:

Step 1: Create a CSV File

1. **Open Excel or any spreadsheet application**.
2. Create a new sheet and enter your data. For example:

clientName	clientEmail
John Doe	john@example.com
Jane Smith	jane@example.com

3. **Save the file as a CSV**:
   - Click on **File** > **Save As**.

- Choose the location where you want to save it.
- Select **CSV (Comma delimited) (*.csv)** from the file type dropdown.
- Name the file (e.g., `clients.csv`) and click **Save**.

Step 2: Set Up Your Postman Collection

1. **Open Postman**.
2. Create or open the collection where you want to add your requests.
3. Ensure your requests (e.g., for creating clients) are set up properly to accept data from a CSV file.

Step 3: Mark Requests for Collection Runner

1. Click on the collection you want to run.
2. Ensure the requests you want to run with the CSV data are marked properly. You can do this by checking the request settings.

Step 4: Run Collection with CSV Data

1. Click on the **Runner** icon (the triangle play button) in the top left corner of Postman.
2. Select your collection from the left sidebar.
3. Click on the **"Select File"** button under the "Data" section.
4. Choose the CSV file (`clients.csv`) you created earlier.
5. Postman will display the number of iterations based on the number of rows in your CSV.

Step 5: Start the Collection Run

1. Click the **"Run"** button to start the execution.
2. Monitor the progress in the Runner. You should see each request being sent with data from the CSV file.

Step 6: Review Results

Once the run is complete, you can check the results for each request in the Runner, including the status and any response data.

	A	B
1	name	job
2	morpheus	leader
3		
4		